

Research Article

A Chaotic Parallel Artificial Fish Swarm Algorithm for Water Quality Monitoring Sensor Networks 3D Coverage Optimization

Jie Zhou , Guohong Qi, and Changzheng Liu 

College of Information Science and Technology, Shihezi University, Shihezi, China

Correspondence should be addressed to Jie Zhou; jiezhou@shzu.edu.cn and Changzheng Liu; 1823982150@qq.com

Received 18 January 2021; Revised 30 January 2021; Accepted 8 February 2021; Published 26 February 2021

Academic Editor: Bin Gao

Copyright © 2021 Jie Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, the increasingly severe water pollution problem encouraged researchers to optimize water quality monitoring sensor networks (WQMSNs) by creating new underwater sensor coverage algorithms. Since the sensor is limited by the monitoring range and the number of targets, optimizing the 3D target coverage of heterogeneous multisensors is essential to maximize the 3D target coverage rate of the monitored waters. To enhance the target coverage rate, the target allocation needs to be searched in all possible combinations. To optimize the 3D coverage of underwater targets, this research proposes a chaotic parallel artificial fish swarm algorithm (CPAFSA). CPAFSA uses chaotic selection to initialize parameters and integrates the global search capabilities of parallel operators. It also applies the elite selection which effectively avoiding local optimization and solving the problem of 3D target coverage. Ultimately, CPAFSA is compared with genetic algorithm (GA) and particle swarm optimization (PSO). The results of the simulation experiment demonstrated the excellent performance of CPAFSA in achieving underwater 3D target coverage.

1. Introduction

In recent years, with economic development and population growth, environmental pollution, especially water pollution, has become increasingly serious. The monitoring of water quality is an important means to prevent and control pollution, so researchers gradually set their sights on real-time water quality monitoring through wireless sensors [1–3]. Since range and target number are the constraints of sensor monitoring, it is necessary to optimize the 3D coverage of heterogeneous multisensors to maximize the target coverage rate of water quality monitoring sensor networks (WQMSNs). However, the optimal coverage of heterogeneous multisensor is an NP-hard problem. The exhaustive method is considered to be a possible way to solve this problem, but its computational complexity is too high to be suitable for actual real-time applications. Most of the research on sensor coverage involves heuristic sensor coverage algorithms [4]. In previous studies, scholars have proposed many optimization algorithms to solve wireless sensor network (WSNs) problems, including genetic algorithm (GA) and particle swarm optimization (PSO) [5–11].

Recently, more and more scholars have proposed various optimization algorithms for sensor optimization [12–16]. The application direction of the optimization algorithm on the sensor has also become broader [17–20].

Water quality monitoring is an important means to prevent and control water pollution. To provide reliable underwater quality monitoring services and data analysis for environmental protection and domestic water use service, many researchers designed and proposed water quality monitoring strategies based on wireless sensor network and Internet of Things.

To improve the coverage of underwater wireless sensor networks (UWSNs) and extend its network lifetime, paper [21] proposed an algorithm combining the virtual force and PSO. The algorithm guides the optimization of the particle swarm, moving the underwater node to a relatively ideal position, thereby accelerating the particle convergence and making the PSO develop toward the target solution. In response to the deployment of wireless sensor nodes, a natural heuristic cuckoo search algorithm was proposed in [22] to find the best deployment position of sensors in a 3D underwater environment. To maximize target coverage with the

least number of sensors, the authors model the deployment of sensors as an optimization problem. Literature [23] studied a method of allocating underwater monitoring coverage resources in sensor networks and proposed positioning and deployment of UWSNs nodes based on GA. The method can maximize the coverage and protection of high-value assets in military applications. In [24], the authors proposed an improved fruit fly optimization algorithm to solve the 3D underwater sensor network coverage optimization problem and designed a 3D space-based network coverage method. The algorithm uses the behavior of fruit flies' preying to optimize global optimal monitoring. This algorithm can quickly obtain the deployment position of sensor nodes, thereby solving the problem of 3D coverage deployment of wireless sensor nodes. In [25], to solve the optimization problem of node redeployment coverage in UWSNs, an underwater sensor network redeployment algorithm based on wolf-pack search technology was proposed. They use sensor nodes to ensure coverage and avoid nodes appearing prematurely.

Research on merging adaptive theory and parallel theory has been carried out since the late 1990s [26]. Developed at the beginning of the 21st century, the artificial fish swarm algorithm (AFSA) is an evolutionary optimization algorithm that tries to find the best solution of optimizing problems by stochastic rules, and it explores the problem region with a probabilistic policy [27]. The theoretical foundations of AFSA were presented by [28]. A chaotic parallel artificial fish swarm algorithm (CPAFSA) is presented for the underwater 3D target coverage problem in this paper. A model of target coverage and monitoring is proposed to maximize the coverage of underwater 3D targets. And, the elite operator is used when updating the individual artificial fish to ensure a better evolution of the fish swarm. The strategy mixes the merits of a parallel selection and a chaotic operator to enhance the global explore capacity. Then, an adaptive adjustment method is used to obtain better experimental results while avoiding local optima.

To illustrate the advantages of CPAFSA in maximizing the underwater 3D target coverage area by WQMSNs, GA and PSO are used for comparisons. On the one hand, compared with genetic algorithms, CPAFSA uses parallel operation to improve optimization capabilities. On the other hand, CPAFSA can overcome the premature problem of traditional genetic algorithms by using chaotic operator. Through these new operations, CPAFSA becomes a suitable global optimization method to find the optimal solution without falling into the local optimum. CPAFSA can create a feasible solution to the underwater 3D target coverage problem of maximizing the underwater 3D target coverage rate of heterogeneous sensors within an acceptable time.

The results, which are simulated based on CPAFSA, GA, and PSO, show that CPAFSA develops a good solution for achieving a higher underwater 3D target coverage rate. CPAFSA has improved the performance of underwater 3D target coverage by combining parallel adjustment and chaotic optimization. It also helps to avoid local optimal.

The structure of this paper is as follows. Section 2 elaborates on the underwater 3D target coverage and surveillance and, then, introduces its target coverage and surveillance

model. CPAFSA is used to extend the performance of underwater 3D target coverage monitoring problems in Section 3. Section 4 shows the results of the simulation experiment and discusses the significance of CPAFSA. Then, Section 5 concludes.

2. Underwater 3D Target Coverage and Monitoring Model

The deployment problem of WQMSNs is directly related to the optimal configuration of its communication bandwidth, node power, analysis and calculation capabilities, and other restricted resources. It also affects the quality of communication, monitoring, and perception services to a large extent. How to ensure the target coverage of the monitoring area under the premise of the limited number of sensors and limited monitoring capabilities is a key issue that determines monitoring performance. This section will discuss the mathematical model of underwater 3D target coverage and monitoring by WQMSNs.

Firstly, in real-time monitoring of underwater targets, since the sensing capabilities of sensors are limited, usually sensors can only perceive a limited number of targets in their monitoring area, and each target needs to be monitored by multiple heterogeneous sensors. Suppose there are X target points that need to be monitored in a piece of water. These target points need to be covered and monitored by underwater wireless sensors, and the number of sensors is Y . Equation (1) represents the coverage relationship between each sensor and each target of WQMSNs.

$$FG = \begin{bmatrix} fg_{1,1} & fg_{1,2} & \cdots & fg_{1,Y-1} & fg_{1,Y} \\ fg_{2,1} & fg_{2,2} & \cdots & fg_{2,Y-1} & fg_{2,Y} \\ \vdots & \vdots & fg_{x,y} & \vdots & \vdots \\ fg_{X-1,1} & fg_{X-1,2} & \cdots & fg_{X-1,Y-1} & fg_{X-1,Y} \\ fg_{X,1} & fg_{X,2} & \cdots & fg_{X,Y-1} & fg_{X,Y} \end{bmatrix} \cdot (fg_{x,y} \in \{0, 1\}). \quad (1)$$

FG represents the coverage relationship matrix between the residual chlorine sensor node and the target node, and $fg_{x,y}$ represents the coverage relationship between the y residual chlorine sensor node and the monitoring target x .

Equation (1) indicates the coverage relationship in WQMSNs, that is, $fg_{x,y} = 1$ means that the monitoring target x is within the y sensor node's monitoring area, and $fg_{x,y} = 0$ indicates that the monitoring target x is outside the monitoring area of the y sensor.

Then, due to the limited monitoring capabilities of sensors, only a limited number of target points within the area of sensor nodes can be monitored. Equation (2) is used to express the monitoring relationship between sensor nodes and covered targets in WQMSNs.

$$JC = \begin{bmatrix} jc_{1,1} & jc_{1,2} & \cdots & jc_{1,V-1} & jc_{1,V} \\ jc_{2,1} & jc_{2,2} & \cdots & jc_{2,V-1} & jc_{2,V} \\ \vdots & \vdots & jc_{u,v} & \vdots & \vdots \\ jc_{U-1,1} & jc_{U-1,2} & \cdots & jc_{U-1,V-1} & jc_{U-1,V} \\ jc_{U,1} & jc_{U,2} & \cdots & jc_{U,V-1} & jc_{U,V} \end{bmatrix} \quad (2)$$

$\cdot (jc_{u,v} \in \{0, 1\})$.

Equation (2) points out the monitoring relationship in WQMSNs, $jc_{u,v} = 1$ indicates that the monitoring target u is within the monitoring range of the v sensor node and u is monitored. $jc_{u,v} = 0$ suggests that the monitoring target u is outside the monitoring area of the v sensor, or u is within the monitoring range but it is not monitored.

If a sensor node can only monitor N target points in the 3D coverage area, the constraint condition of the monitoring relationship is shown as

$$\sum_{u=1}^U jc_{u,v} \leq N, v \in V. \quad (3)$$

If each monitored target point must be monitored by at least M sensors and a sensor can merely monitor N target points in the coverage area, the mathematical model of underwater 3D target coverage for monitoring more target points in WQMSNs is

$$jc_{u,v} \leq fg_{u,v}. \quad (4)$$

Equation (4) represents the relationship between the monitoring matrix and the coverage matrix. Only when the sensor covers the target point, the corresponding target point in the monitoring matrix may be monitored by the sensor. Considering that the limitation of the number of targets monitored by the sensor, it is also possible that the target points are covered but not monitored.

$$W(u) = \sum_{v=1}^V jc_{u,v}, u \in U. \quad (5)$$

$$WM(u) = \begin{cases} 0 & W(u) < M \\ 1 & W(u) \geq M \end{cases}. \quad (6)$$

Equation (5) uses W to store the number of target points monitored by the sensor and, then, determines whether the point is effectively monitored according to the restriction condition (6).

The above clarifies the optimization direction by establishing a mathematical model for underwater sensor coverage and monitoring target points. The next section will design and implement a CPAFSA that expands the underwater 3D coverage and monitoring rate of sensors based on the model.

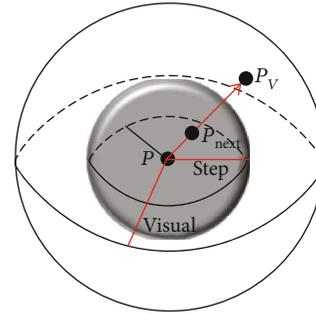


FIGURE 1: Simulated artificial fish vision.

3. CPAFSA for Maximizing Underwater 3D Coverage and Monitoring Rate in WQMSNs

The basis of the artificial intelligence model based on biological behavior is the bottom-up design method. This model designs the behavioral perception of a single entity and then places the individual or group in the environment so that it can propose solutions to problems in the interaction with the environment. Individuals usually do not have advanced intelligence, but they can show advanced intelligence during group activities. This phenomenon is called swarm intelligence. Individuals with social characteristics can produce group intelligence when they cooperate in certain activities, such as fish swarm.

Artificial fish swarm is an abstraction of a biological fish swarm, which simulates the characteristic behavior of biological fish and their response to environmental stimuli. Individual artificial fish can receive environmental information through vision and respond accordingly, and the action of the individual artificial fish will also affect the other artificial fish individuals. Artificial fish's perception of the environment is realized through vision. But the visual system of biological fish is very complicated, so the concept of the visual field is adopted when simulating artificial fish vision. Figure 1 simulates the relationship between the field of view of an artificial fish and its step length. The current state of the individual fish is set to $P = (p_1, p_2, p_3, \dots, p_n)$; artificial fish's field of vision is *Visual* and chaotically selects a state in its *Version* at a certain time $P_v = (p_{1v}, p_{2v}, p_{3v}, \dots, p_{nv})$; if the state P_v is better than P , the fish will take a random step towards P_v to reach state P_{next} . Otherwise, continue to try to randomly select other states in its *Version*. The more the individual fish try, the better they can understand the environmental information in the field of view. This helps to make correct behavior decisions. But the actual search behavior of biological fish cannot increase indefinitely, the number of inspections of artificial fish is also limited. Preserving the uncertain local optimization of artificial fish is conducive to artificial fish searching for the global optimum.

P in Figure 1 represents the current state of the individual fish, including two parameters: visual field of view and step length. Among them, *Visual* is used for the perception of other fish individuals within the field of view, and *Step* is the current moveable range of P . In the 3D search space, $X = (\text{abscissa}, \text{ordinate}, \text{vertical})$. And the fish

swarm wants to find the area with the highest concentration of food, which is also called fitness here, expressed as $Y = f(X)$. The spatial distance between individual fish is expressed as $d_{i,j} = \|X_i - X_j\|$. Besides, fish swarm parameters also include the maximum number of attempts in the fish swarm for preying $TryNumber$, the crowdedness degree of the fish swarm δ , and the number of fish within the field of view of a single fish that is n_f .

In the AFSA, there are mainly four kinds of behaviors: preying behavior, swarming behavior, following behavior, and random behavior to simulate biological fish swarm. Preying behavior is the basic behavior of artificial fish. Artificial fish use the perception parameter $Visual$ of the environment to change their position. For different fish state, the food concentration is different. Each fish compares the food concentration at its own location with the randomly selected food concentration in perception $Visual$ and selects the moving direction of the larger food concentration so that the entire fish swarm tends to a high concentration position. Swarming behavior means that each fish moves to the center of the neighboring other artificial fish to ensure that the surrounding areas are all partners to reduce the risk. When the swarming behavior is in progress, the artificial fish in the AFSA are all moving to a place where food is concentrated. Each fish can form a group through grouping behavior and move to a high concentration position together. Following is the behavior of artificial fish chasing the fish with the highest food density among other fish nearby. Due to the nature of the fish swarm toward food and away from natural enemies, when some individuals in the group find food and move in a certain direction, other individuals will follow it. The following reduces the time when artificial fish explore the surrounding environment and also reduces the computing time. Random behavior refers to the behavior when the number of times the artificial fish prey on food reaches the specified maximum number of times, and the food concentration is still not increased. At this moment, the fish randomly moves one step to a certain state in its field of view and uses this state as the next state. Random behavior helps the individual fish to escape from the local optimum. Besides, an artificial fish swarm is mainly affected by three items in the optimization process: bulletin board, behavior evaluation, and iteration termination conditions.

The traditional AFSA has strong global convergence, but it is easy to cause the problem which fish swarms difficult to escape from the local optimum in the later stage of execution. This section proposes CPAFSA. It generates fish swarm sequence through chaotic mapping and then generates chaotic adaptive function coefficients to improve the field of view and step length of fish. So that the artificial fish's field of view and step size change consistent with the parameter requirements of different execution stages of the algorithm. Moreover, to enhance the low accuracy of the AFSA optimal solution, parallel technology is used to improve it. We also use elite operators to make the fish swarm search more efficient. Section 5 compares and analyzes the optimization results of the underwater 3D coverage area of CPAFSA, PSO, and GA.

The execution of the CPAFSA algorithm is a process of intelligent optimization. The main steps are as follows:

Step 1: initialize Q individual artificial fish, moving step $Step$, visual field $Visual$, number of attempts $TryNumber$, congestion factor δ , current number of iterations = 0, maximum number of iterations Max , probability factor α ($0 < \alpha < 1$), constant coefficient S

Step 2: calculate the fitness function of Q artificial fish, also called food concentration, obtain the best sensor allocation plan, and assign the fitness function to the bulletin board BB

Step 3: if the center position P_c is better than the current artificial fish position P_i and is not too crowded, set $Step = Rand * \|X_c - X_i\|$ and move one step to the center position; otherwise, go to Step5

Step 4: if the optimal fish P_{max} in the current artificial fish P_i 's perception range is better than artificial fish in the current state, and not too crowded, set $Step = Rand * \|X_{max} - X_i\|$ to move one step to the optimal fish; otherwise, switch to Step 5

Step 5: select a random state P_j in the current field of vision of artificial fish P_i , if $\Delta Y = Y_j - Y_i > 0$, set $Step = Rand * \|X_j - X_i\|$, move to the artificial fish one by one random steps. If after try $TryNumber$ times, there is still no result that meets the conditions, a random position is selected as the next position in the field of $Visual$, and a random step is moved to this position. Update the bulletin board

Step 6: if iteration $< Max$, then set iteration = iteration + 1 and switch to Step 3; if iteration $\geq Max$, switch to Step7

Step 7: output the optimal plan information of the bulletin board

Figure 2 shows the process of the CPAFSA algorithm in solving the underwater 3D coverage problem of WQMSNs.

CPAFSA largely depends on the coding method that solves the underwater 3D coverage and monitoring problem of WQMSNs. Because CPAFSA is an optimized algorithm for simulating biological fish swarm, individual artificial fish are the solution to the sensor coverage problem. Each individual fish is an independent solution to the monitoring optimization problem of WQMSNs. The information needed to construct artificial fish collaboration can be expressed as $\partial = [\text{fish swarm size, prey, follow, swarm}]$. The individual artificial fish is coded as a one-dimensional array $P = (p_1, p_2, p_3, \dots, p_n)$, and it has a 3D coordinate. Determine the swimming direction of the individual fish by the target value corresponding to each element in the array, so as to maximize the underwater 3D coverage. The coding scheme of the one-dimensional array actually limits the fish; then, it can reflect the 3D coverage of the water quality monitoring sensor network. In CPAFSA, a swarm of fish is composed of a certain number of individual fish, and this fish swarm represents all the solutions to the underwater 3D coverage problem of the sensor.

The other parameters of CPAFSA are set as follows.

Under the premise of system resources and running time permitting, using a large population can improve the accuracy of the optimal solution and enhance the ability of the fish swarm to jump out of the local optimal.

The artificial fish's field of view mainly affects its foraging situation. To understand the information of the optimization

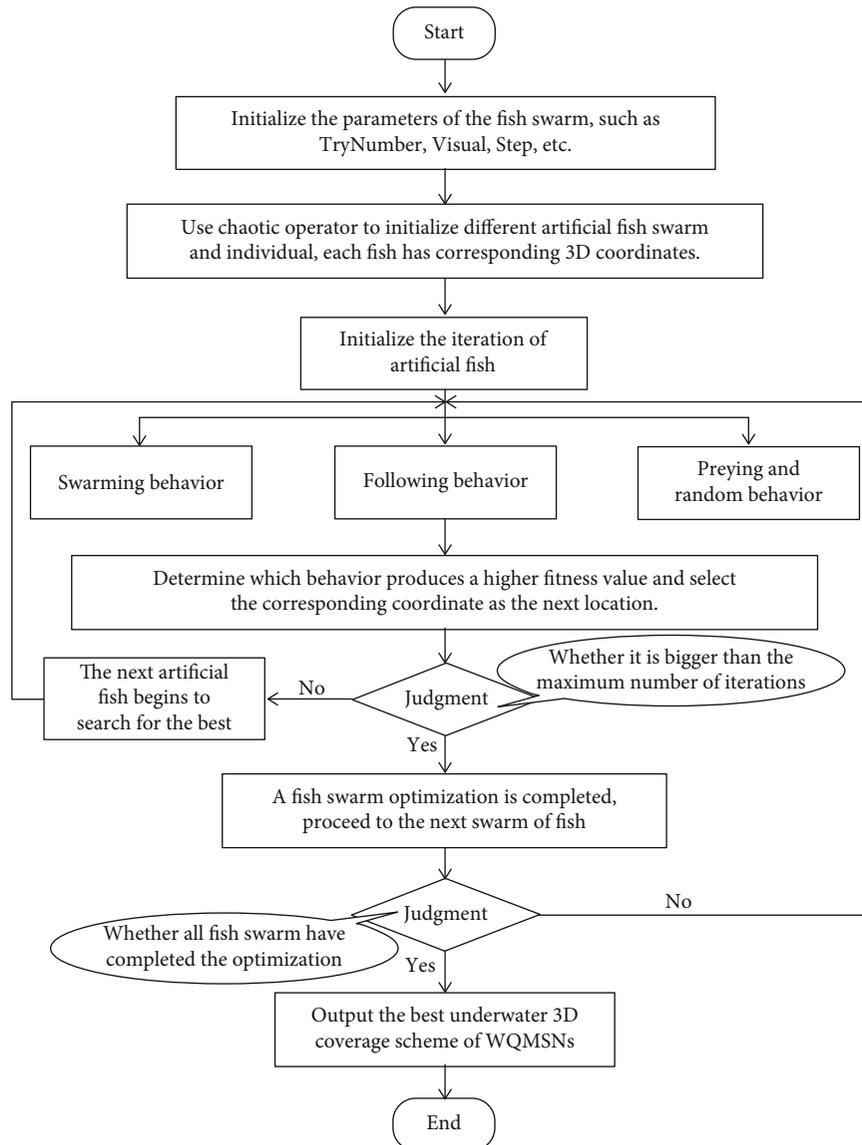


FIGURE 2: Process of the CPAFSA.

space as much as possible, a larger field of view should be used, but a larger field of view will cause the problem of reduced optimization accuracy.

When setting the step length of the artificial fish, it is necessary to consider the size of the fish field of view. As the field of view increases, the step length should also increase accordingly; otherwise, the convergence speed will be slower, but the optimal solution accuracy will decrease. The smaller the step size set by the artificial fish, the lower the convergence speed of the algorithm, which may fall into the local optimum.

The congestion factor parameter is integrated according to the literature [27].

Too many *TryNumber* attempts will cause the artificial fish to be trapped by local extremes, resulting in premature algorithm maturity and increased optimization time. Fewer preying times will reduce the probability of individual artificial fish preying successfully, causing artificial fish to perform

more random behaviors, which is not conducive for algorithm convergence. Generally, the preying attempts of artificial fish do not exceed 100 times, and usually 5–50 times. When the local optimum is not significant and the algorithm complexity is not high, increasing the number of preying is an effective means to improve the convergence efficiency of the algorithm.

The bulletin board records the best solution to the WOMSNs 3D monitoring problem in the current artificial fish swarm. When there is a record of the best solution on the bulletin board, the individual artificial fish in this iterative process needs to compare the solution at its location with the best solution on the bulletin board. If the solution represented by the fish is better than the best solution recorded on the bulletin board at this time, the value of the best solution of the artificial fish on the board will be rewritten after the end of this iteration process, so that the bulletin board stores the optimal solution in the optimization process. The

The swarming behavior of CPAFSA

```

CPAFSA swarm () {
  Xc = 0; nf = 0;
  for (j = 0; j < fishnum; j++) {
    if (di,j < Visual)
      nf++; Xc = Xc + Xj;
    Xc =  $\frac{X_c}{n_f}$ ;
  }
  if (Yc/nf > δ * Yi)
    Xnext = Xi +  $\frac{X_c - X_i}{\|X_c - X_i\|} * Rand() * Step$ ;
  else
    CPAFSA prey ();
}

```

FIGURE 3: Pseudocode of swarming behavior.

historical optimal solution was produced. $BB = 0$ when initializing the bulletin board.

Assume that the state of the individual artificial fish in CPAFSA at this moment is X_i and its fitness function is Y_i , each individual fish perceives the number of other individuals n_f according to its visual field *Visual* and calculates the average state X_c as the center point

$$X_c = \frac{X_1 + X_2 + \dots + X_{n_f}}{n_f}. \quad (7)$$

Define the fitness value of the center position as Y_c/n_f . If $Y_c/n_f > \delta * Y_i$, it means that the center has more food and is not too crowded, so it will randomly move one step toward the center. Otherwise, the individual fish will perform prey behavior. The mathematical expression of clustering behavior is shown in equations (8) and (9).

$$X_{next} = X_i + \frac{X_c - X_i}{\|X_c - X_i\|} * Rand() * Step, \frac{Y_c}{n_f} > \delta * Y_i. \quad (8)$$

$$X_{next} = prey(X_i). \quad (9)$$

The pseudocode of CPAFSA's swarming behavior is shown in Figure 3.

The following behavior simulates the process of a swarm of biological fish moving toward the food source. When a certain fish finds food, the surrounding fish will follow it to swim toward the food. In artificial fish swarm of CPAFSA, this behavior is imitated as the current artificial fish X_i searching for the position X_{max} of the fish with the largest fitness value in its *Version*. When there is X_{max} , and the corresponding fitness value $Y_{max}/n_f > \delta * Y_i$ means that the location is not currently crowded, the artificial fish will move to the optimal direction by a random step. Otherwise, the individual artificial fish will prey. Equations (10) and (9) express the mathematical following behavior.

$$X_{next} = X_i + \frac{X_{max} - X_i}{\|X_{max} - X_i\|} * Rand() * Step, Y_{max}/n_f > \delta * Y_i. \quad (10)$$

The following behavior of CPAFSA

```

CPAFSA follow () {
  Ymax = 0; nf = 0;
  for (j = 0; j < fishnum; j++) {
    if (di,j < Visual && Ymax < Yj) {
      Ymax = Yj; Xmax = Xj;
    }
    if (di,j < Visual) nf++;
  }
  if (Ymax/nf > δ * Yi)
    Xnext = Xi +  $\frac{X_{max} - X_i}{\|X_{max} - X_i\|} * Rand() * Step$ ;
  else
    CPAFSA prey ();
}

```

FIGURE 4: Pseudocode of following behavior.

The preying and random behaviors of CPAFSA

```

CPAFSA prey () {
  for (j = 0; j < fishnum; j++) {
    Xj = Xi + Rand() * Visual;
    if (Yi < Yj)
      Xnext = Xi +  $\frac{X_j - X_i}{\|X_j - X_i\|} * Rand() * Step$ ;
    else
      Xnext = Xi + Rand() * Step;
  }
}

```

FIGURE 5: Pseudocode of preying and random behaviors.

Figure 4 shows the pseudocode of CPAFSA's following behavior.

Preying is the instinct of animals and the basis of biological evolution. When biological fish find an area with a higher food concentration, they will instinctively swim towards there. This behavior is manifested in the artificial fish swarm of CPAFSA. The fish individual randomly selects the state position X_j within its perception range *Visual*, and obtains the fitness value Y_j at this moment. If $Y_j > Y_i$, move the individual fish to the X_j position by a random step. If $Y_j < Y_i$, the individual fish randomly selects a state position in the *Visual* again for judgment. When the iteration reaches the maximum number of attempts *TryNumber* and still does not find a qualified Y_j , the artificial fish will execute into a random state, that is, randomly select a state in the field of *Version* and move to that state, thereby avoiding local optimal. Preying and random behaviors are expressed in mathematical language as equations (11)–(13).

$$X_j = X_i + Rand() * Visual. \quad (11)$$

$$X_{next} = X_i + \frac{X_j - X_i}{\|X_j - X_i\|} * Rand() * Step, Y_j > Y_i. \quad (12)$$

$$X_{next} = X_i + Rand() * Step, Y_j < Y_i. \quad (13)$$

The preying and random behaviors of CPAFSA uses pseudocode as Figure 5.

TABLE 1: Parameters settings of CPAFSA.

	Generations	Population	Visual	Step	Congestion factor	Preying attempts
CPAFSA	100	20	12	20	0.618	10

TABLE 2: Parameters settings of PSO.

	Generations	Population	Maximum speed	Individual and social learning factors
PSO	100	20	1	2

TABLE 3: Parameters settings of GA.

	Generations	Population	Mutation probability
GA	100	20	0.01

As a universal phenomenon in nonlinear systems, chaos has diversity and multiscale. These characteristics make chaos theory have great application potential in the field of algorithm optimization. For the nonlinear engineering optimization problem of underwater 3D sensor coverage problem in WQMSNs, CPAFSA uses chaotic mapping to determine the initial coordinates of underwater sensors and targets, which saves workload and improves randomness. Then, CPAFSA applied chaos search to the individual swimming process of individual artificial fish to enhance the search ability of individual fish and the ability of local optimization.

CPAFSA solves the WQMSNs problem from the angle of the fish swarm, and adding parallel operators can greatly improve the solution quality and accuracy of the algorithm. Parallel models include the fine-grained model, master-slave model, and coarse-grained model. The fine-grained model is mainly used in large-scale computer systems. It can maximize the parallelization capability of the algorithm but has very high hardware requirements. The master-slave model has a master processor and multiple slave processors. In this model, the global processing operations of the fish swarm are executed in the main processor, and the following behavior, swarming behavior, random behavior, and preying behavior are executed in the secondary processor. The coarse-grained model is an overall parallel model. The model divides the entire large fish swarm into multiple scattered fish swarms. Each scattered fish swarm evolves independently. After a certain algebraic evolution, compare different fish swarms and copy excellent fish individuals to other fish swarms, thereby improving the search ability of the algorithm in the local environment. CPAFSA uses a coarse-grained parallel model to improve the overall optimization capabilities of WQMSNs in this paper.

4. Results and Discussion

In this section, we compare the CPAFSA, PSO, and GA algorithms and analyze the performance of the algorithms in solving WQMSNs underwater 3D coverage and monitoring

optimization. Then, the advantages of CPAFSA in solving this problem were demonstrated through simulation experiments with different parameter settings. The three algorithms judge the pros and cons of the optimization effect through a common fitness function.

In the general parameter setting, the monitoring range of the sensor is $400 \times 400 \times 400 \text{ m}^3$. The number of sensors is 35 and the number of monitored targets is 100. The positions of monitored targets and sensor nodes are randomly distributed due to the chaos operator. At the same time, each target point needs to be monitored by 3 sensors, and one sensor can only monitor 4 target points in the coverage area. Besides, CPAFSA, PSO, and GA use the same algebra and scale for comparison. Tables 1–3 show the parameter values of CPAFSA, PSO, and GA.

In CPAFSA, the number of individual fish in the artificial fish school is set to 40. The setting of step length and field of view parameters has an important impact on the performance of this algorithm. According to the CPAFSA optimized for underwater 3D coverage problem in WQMSNs in Section 4, the field of view and step length parameters are set. And based on the comprehensive evaluation of relevant literature, the crowding factor and the maximum number of swarming are set [29–31]. Table 1 shows these parameters.

PSO is a swarm optimization algorithm inspired by the phenomenon of bird swarm preying. It realizes the calculation of spatial solutions through collaboration and imitation between individuals. The particle has a velocity vector feature, which determines the state of the bird's flight. The particles follow the best particles in the current state to search in space and optimize the initialized random particles to find the best result through iteration. In each generation, the process of updating particles is carried out by tracking two extreme values. Extreme value 1 is the historical optimal solution of the particle itself, and extreme value 2 is the optimal solution of the overall particle. Each particle gathers at a certain speed to the best position in its own history and the best position in the neighborhood history to realize the evolution of candidate solutions. Table 2 shows the parameters of PSO.

GA is an optimization algorithm that combines genetic theory with computer technology. Many terms in this algorithm are derived from natural evolution theory. The chromosomes carry the genetic material of the organism, which controls the traits of the organism, and the gene is the functional and combined representation of the genetic material, and the value of a gene is called an allele. A certain number of genes make up a chromosome. The position on the chromosome is called a locus. The combination of genes and locus determines the characteristics of the chromosome, and the traits of an organism are external manifestations. Genetic operators such as selection probability and crossover

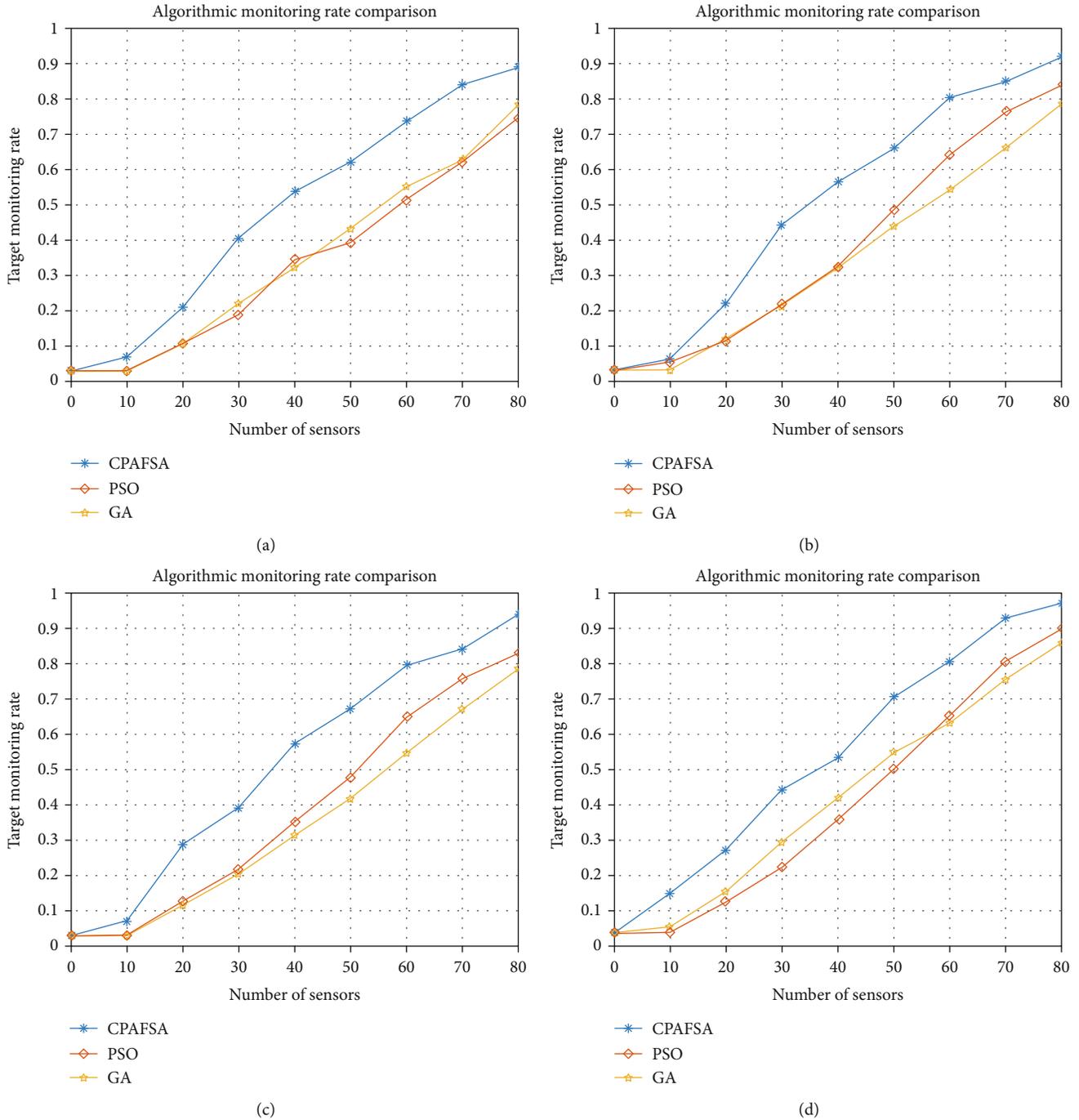


FIGURE 6: The change curve of the target ratio of successful monitoring as the number of sensor nodes increases: (a) radius of 120 meters; (b) radius of 140 meters; (c) radius of 160 meters; (d) radius of 180 meters.

probability need to be set in GA. Moreover, in the process of running the algorithm, GA also needs to set parameters such as the length of the individual code string and the group size. Table 3 shows the parameter settings of GA.

Figure 6 shows the change in the number of targets successfully monitored underwater caused by the increase in the number of sensor nodes. (a), (b), (c), and (d), respectively, represent the cases where the sensor node monitoring radius is 120 meters, 140 meters, 160 meters, and 180 meters. It can be seen from the figure that as the number of sensors

increases, the proportion of targets successfully monitored for GA and PSO has increased, but they are always lower than CPAFSA. CPAFSA optimizes through parallel comparison, which increases the search range compared to the previous two algorithms. Meanwhile, CPAFSA will go through the update process of the optimal solution in each iteration process and can monitor more targets more efficiently. According to Figure 3, Tables 4 and 5, respectively, show the percentage increase of target nodes successfully monitored by CPAFSA compared to GA and PSO. It can be seen

TABLE 4: The percentage increase of CPAFSA's successful monitoring targets compared to PSO.

Number of sensors	Monitoring radius of 120 meters	Monitoring radius of 140 meters	Monitoring radius of 160 meters	Monitoring radius of 180 meters
10	4.20%	1.00%	4.20%	11.00%
20	10.20%	10.60%	16.40%	14.60%
30	21.80%	22.60%	17.40%	21.80%
40	19.40%	23.80%	22.00%	17.40%
50	23.00%	17.40%	19.60%	20.40%
60	22.20%	16.20%	14.40%	15.20%
70	22.00%	8.60%	8.40%	12.40%
80	14.40%	8.00%	11.20%	7.20%

TABLE 5: The percentage increase of CPAFSA's successful monitoring targets compared to GA.

Number of sensors	Monitoring radius of 120 meters	Monitoring radius of 140 meters	Monitoring radius of 160 meters	Monitoring radius of 180 meters
10	4.00%	3.20%	4.20%	9.40%
20	10.20%	9.80%	17.40%	11.60%
30	18.40%	23.00%	18.80%	14.60%
40	21.60%	24.20%	25.80%	11.20%
50	18.80%	22.00%	25.40%	15.80%
60	18.60%	26.00%	24.60%	17.20%
70	21.20%	18.80%	17.00%	17.60%
80	10.60%	13.40%	15.40%	11.00%

from the table that the percentage of CPAFSA successfully monitored targets are always 1.00% to 23.80% higher than that of PSO, and 3.20% to 25.80% higher than that of GA. CPAFSA can significantly improve the monitoring effect.

Figure 6 also shows that when the number of sensors increases, the targets successfully monitored by CPAFSA generally increase more than the other two algorithms. The reason is that as the number of sensors and targets continues to grow, the complexity of the 3D coverage of underwater sensors continues to increase. The traditional GA and PSO are prone to fall into premature convergence, while CPAFSA has a parallel operator added to the code, which can perform a broad global optimization. In addition, the clustering and foraging search can transform the global search into the local search, which improves the balance between coarse search and fine search, also improves the speed of algorithm convergence, and avoids falling into the local extremum.

Figure 7 shows the number of targets successfully monitored when the number of target points is 100, the number of sensors is 80 and 100, and the corresponding radius is 130 meters and 150 meters, respectively. The underwater 3D coverage monitoring method is based on CPAFSA, GA, and PSO in WQMSNs. The definition of target success monitoring rate is the percentage of the number of targets successfully monitored to the total number of targets within the sensor coverage. To test the actual effect of the algorithm under different types of sensors, we set the number of sensors in (a) to 80, the radius to 130 meters. Set the number of sensors in (b) to 80, and the radius to 150 meters. Then, the number of sensors in (c) is set to 100, the radius is set to

130 meters. And the number of sensors in (d) is set to 100; the radius is set to 150 meters. The results of the simulation experiment explain that as generations of the algorithm increases, the PSO-based target coverage monitoring rate quickly stabilizes, which indicates that the PSO will quickly reach a local extreme and it is difficult to jump out of the optimization stage. The coverage rate of the GA-based target coverage method is steadily increasing. However, because GA uses a fixed zero-one code, it is difficult to introduce new genes when the mutation rate is low, and it is easy to fall into premature maturity. When the probability of mutation is high, individual adaptability will fluctuate greatly, and the target success monitoring rate will also be high. It is difficult to improve. The single fish in CPAFSA adopts zero-one coding based on chaos to avoid evolutionary stagnation caused by local optimization. It can be seen from the figure that CPAFSA can always monitor more targets than PSO and GA regardless of the same number of sensors but different radii or the same number of sensors and different radii.

Figure 8 shows the percentage of successful surveillance targets based on CPAFSA, GA, and PSO in WQMSNs. To determine the utilization performance of the algorithm under different conditions, the number of sensors in (a) is set to 90, and the radius is set to 190 meters. Set the number of sensors in (b) to 80 and the radius to 200 meters. The number of sensors in (c) is set to 70, and the radius is set to 210 meters. The number of sensors in (d) is set to 60, and the radius is set to 220 meters. It can be seen from the figure that the percentages of successfully monitoring targets based on CPAFSA in (a), (b), (c), and (d) are all above

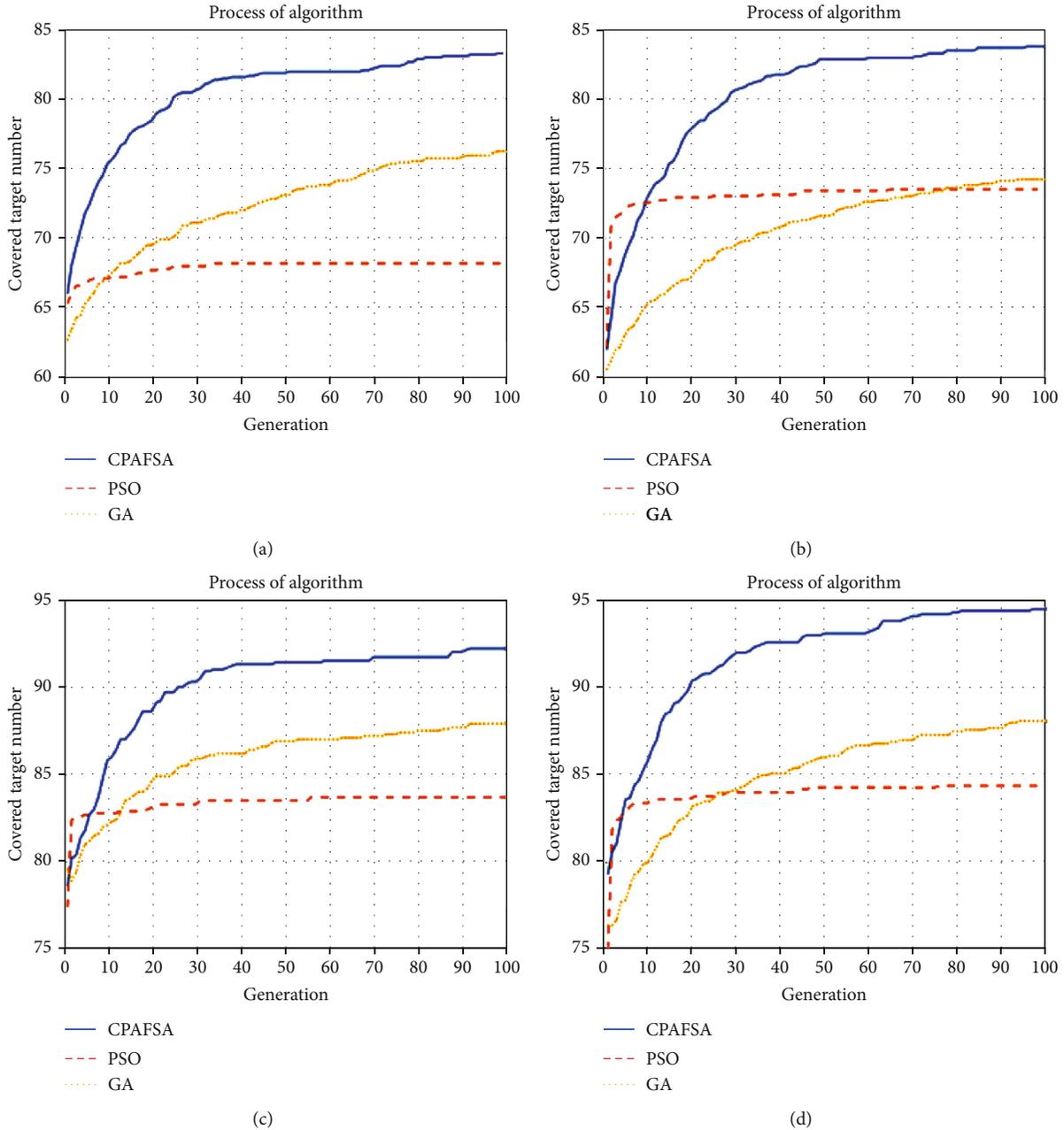


FIGURE 7: The number of successfully monitored targets varies with algorithm iterations: (a) 80 sensors, 130 meters radius; (b) 80 sensors, 150 meters radius; (c) 100 sensors, 130 meters radius; (d) 100 sensors, 150 meters radius.

90%, and the percentages of PSO and GA successfully monitoring targets decrease as the number of sensors decreases. Under certain conditions, compared to PSO and GA, CPAFSA can use sensors for more effective coverage and monitoring in WQMSNs.

To verify the performance of CPAFSA, this section compares this algorithm with PSO and GA in terms of the number and proportion of successfully monitored targets. The simulation results show that the newly proposed CPAFSA has stronger global optimization capabilities than PSO and GA in solving the underwater 3D coverage optimization problem in WQMSN, and it has not converged prematurely.

5. Conclusion

Aiming at the optimal coverage of the water quality monitoring sensor networks, this paper proposed a new chaotic parallel artificial fish swarm algorithm. Before the algorithm design, an underwater 3D coverage model was established to solve the sensor coverage and monitoring problem. This model optimizes the 3D coverage of underwater wireless sensors to target points when the sensors are limited. To prevent AFSA from converging too fast in the optimization process and falling into a local optimum, CPAFSA uses a chaos operator to enhance the randomness of CPAFSA's initial setting.

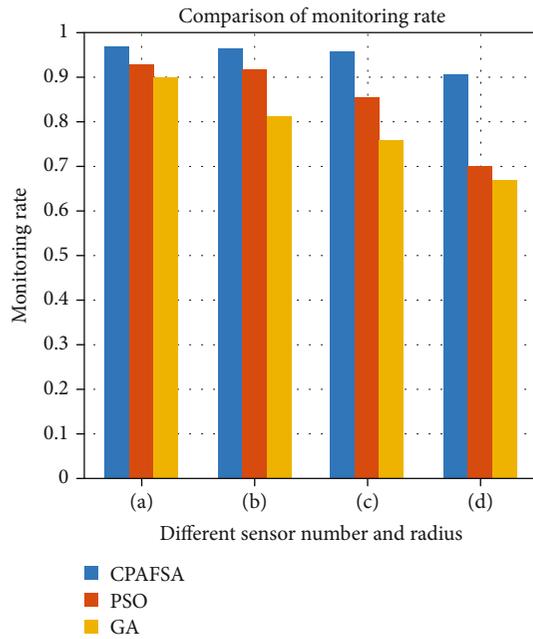


FIGURE 8: Percentage of successfully monitored targets: (a) 90 sensors with a radius of 190 meters; (b) 80 sensors with a radius of 200 meters; (c) 70 sensors with a radius of 210 meters; (d) 60 sensors with a radius of 220 meters.

And this algorithm enhances the ability of global fish swarm optimization through parallel strategies; thus, AFSA is optimized and the shortcomings of insufficient precision are solved. Besides, this algorithm also combines the elite operator and the adaptive operator in the optimization process, thereby improving the optimization efficiency of a single artificial fish and preventing the ineffective search.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was funded by the Corps innovative talents plan, grant number 2020CB001; the project of Youth and Middle-aged Scientific and Technological Innovation Leading Talents Program of the Corps, grant number 2018CB006; the China Postdoctoral Science Foundation, grant number 220531; the Funding Project for High Level Talents Research in Shihezi University, grant number RCZK2018C38; and the Project of Shihezi University, grant number ZZZC201915B.

References

- [1] H. Jafari, T. Rajaei, and S. Nazif, "An investigation of the possible scenarios for the optimal locating of quality sensors in the

water distribution networks with uncertain contamination," *Journal of Water and Health*, vol. 18, no. 5, pp. 704–721, 2020.

- [2] K. S. Adu-Manu, F. A. Katsriku, J.-D. Abdulai, and F. Engmann, "Smart river monitoring using wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8897126, 19 pages, 2020.
- [3] S. Cheng and C. Li, "Research and design of water quality monitoring system based on automatic monitoring of mobile aeration equipment," in *AIP conference proceedings*, Chongqing, China, 2019no. 1, Article ID 020049.
- [4] A. B. S. Yıldız, N. Pholdee, S. Bureerat, A. R. Yıldız, and S. M. Sait, "Sine-cosine optimization algorithm for the conceptual design of automobile components," *Materials Testing*, vol. 62, no. 7, pp. 744–748, 2020.
- [5] S. E. Bouzid, Y. Seresstou, K. Raoof, M. N. Omri, M. Mbarki, and C. Dridi, "MOONGA: multi-objective optimization of wireless network approach based on genetic algorithm," *IEEE Access*, vol. 8, pp. 105793–105814, 2020.
- [6] M. Elhoseny, A. Tharwat, A. Farouk, and A. E. Hassanien, "K-coverage model based on genetic algorithm to extend WSN lifetime," *IEEE Sensors Letters*, vol. 1, no. 4, article 7500404, pp. 1–4, 2017.
- [7] J. Li, Z. Luo, and J. Xiao, "A hybrid genetic algorithm with bidirectional mutation for maximizing lifetime of heterogeneous wireless sensor networks," *IEEE Access*, vol. 8, pp. 72261–72274, 2020.
- [8] J. Wang, Y. Cao, B. Li, H. J. Kim, and S. Lee, "Particle swarm optimization based clustering algorithm with mobile sink for WSNs," *Future Generation Computer Systems*, vol. 76, pp. 452–457, 2016.
- [9] S. C. Manju and B. Kumar, "Genetic algorithm-based meta-heuristic for target coverage problem," *IET Wireless Sensor Systems*, vol. 8, no. 4, pp. 170–175, 2018.
- [10] Z. Jiao, L. Zhang, M. Xu, C. Cai, and J. Xiong, "Coverage control algorithm-based adaptive particle swarm optimization and node sleeping in wireless multimedia sensor networks," *IEEE Access*, vol. 7, pp. 170096–170105, 2019.
- [11] T. Qasim, M. Zia, Q. A. Minhas et al., "An ant colony optimization based approach for minimum cost coverage on 3-D grid in wireless sensor networks," *IEEE Communications Letters*, vol. 22, no. 6, pp. 1140–1143, 2018.
- [12] N. Panagant, N. Pholdee, S. Bureerat, A. R. Yıldız, and S. M. Sait, "Seagull optimization algorithm for solving real-world design optimization problems," *Materials Testing*, vol. 62, no. 6, pp. 640–644, 2020.
- [13] B. S. Yıldız, A. R. Yıldız, N. Pholdee, S. Bureerat, S. M. Sait, and V. Patel, "The Henry gas solubility optimization algorithm for optimum structural design of automobile brake components," *Materials Testing*, vol. 62, no. 3, pp. 261–264, 2020.
- [14] B. S. Yıldız, "Optimal design of automobile structures using moth-flame optimization algorithm and response surface methodology," *Materials Testing*, vol. 62, no. 4, pp. 371–377, 2020.
- [15] B. S. Yıldız, "The mine blast algorithm for the structural optimization of electrical vehicle components," *Materials Testing*, vol. 62, no. 5, pp. 497–502, 2020.
- [16] B. S. Yıldız, A. R. Yıldız, E. İ. Albak, H. Abderazek, S. M. Sait, and S. Bureerat, "Butterfly optimization algorithm for optimum shape design of automobile suspension components," *Materials Testing*, vol. 62, no. 4, pp. 365–370, 2020.
- [17] H. Abderazek, A. R. Yıldız, and S. Mirjalili, "Comparison of recent optimization algorithms for design optimization

- of a cam-follower mechanism,” *Knowledge-Based Systems*, vol. 191, p. 105237, 2020.
- [18] E. Kurtuluş, A. R. Yıldız, S. M. Sait, and S. Bureerat, “A novel hybrid Harris hawks-simulated annealing algorithm and RBF-based metamodel for design optimization of highway guardrails,” *Materials testing*, vol. 62, no. 3, pp. 251–260, 2020.
- [19] A. R. Yıldız, B. S. Yıldız, S. M. Sait, S. Bureerat, and N. Pholdee, “A new hybrid Harris hawks-Nelder-Mead optimization algorithm for solving design and manufacturing problems,” *Materials Testing*, vol. 61, no. 8, pp. 735–743, 2019.
- [20] F. Hamza, H. Abderazek, S. Lakhdar, D. Ferhat, and A. R. Yıldız, “Optimum design of cam-roller follower mechanism using a new evolutionary algorithm,” *The International Journal of Advanced Manufacturing Technology*, vol. 99, no. 5-8, pp. 1267–1282, 2018.
- [21] Y. Sun, Y. Hu, L. Chen, H. Liu, J. Chen, and B. Lv, “The coverage optimization method for underwater sensor network based on VF-PSO algorithm,” in *2020 Chinese Control And Decision Conference (CCDC)2020 Chinese Control And Decision Conference (CCDC)*, pp. 2008–2013, Hefei, China, 2020.
- [22] D. Arivudainambi, S. Balaji, and T. S. Poorani, “Sensor deployment for target coverage in underwater wireless sensor network,” in *2017 International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*, pp. 1–6, Paris, 2017.
- [23] S. Iyer and D. V. Rao, “Genetic algorithm based optimization technique for underwater sensor network positioning and deployment,” *IEEE.2015 IEEE underwater technology (UT)*, pp. 1–6, Chennai, India, 2015.
- [24] Y. Zhang, M. Wang, J. Liang, H. Zhang, W. Chen, and S. Jiang, “Coverage enhancing of 3D underwater sensor networks based on improved fruit fly optimization algorithm,” *Soft Computing*, vol. 21, no. 20, pp. 6019–6029, 2017.
- [25] P. Jiang, Y. Feng, and F. Wu, “Underwater sensor network redeployment algorithm based on wolf search,” *Sensors*, vol. 16, no. 10, article 1754, 2016.
- [26] C. W. Ou and S. Ranka, “Parallel remapping of adaptive problems,” *Journal of Parallel and Distributed Computing*, vol. 42, no. 2, pp. 109–121, 1997.
- [27] X. L. Li and J. X. Qian, “Studies on artificial fish swarm optimization algorithm based on decomposition and coordination techniques,” *Journal of Circuits and Systems*, vol. 1, pp. 1–6, 2003.
- [28] C.-R. Wang, C.-L. Zhou, and J.-W. Ma, “An improved artificial fish-swarm algorithm and its application in feed-forward neural networks,” in *2005 International conference on machine learning and cybernetics*, vol. 5, pp. 2890–2894, Guangzhou, China, 2005.
- [29] X. Zhou, Z. Wang, D. Li, H. Zhou, Y. Qin, and J. Wang, “Guidance systematic error separation for mobile launch vehicles using artificial fish swarm algorithm,” *IEEE Access*, vol. 7, pp. 31422–31434, 2019.
- [30] L. I. Zhanwu, C. Yizhe, K. O. Yingxin, Y. A. Haiyan, X. U. An, and L. I. You, “Approach to WTA in air combat using IAFSA-IHS algorithm,” *Journal of Systems Engineering and Electronics*, vol. 29, no. 3, pp. 519–529, 2018.
- [31] G. Zhou, Y. Li, Y. C. He, X. Wang, and M. Yu, “Artificial fish swarm based power allocation algorithm for MIMO-OFDM relay underwater acoustic communication,” *Iet Communications*, vol. 12, no. 9, pp. 1079–1085, 2017.